

## Web Applications I – Exam # 1 (deadline 2023-06-25 at 23:59)

# “CMSmall”

FINAL VERSION – Modifications are reported in “red”

Design and implement a web application for a small Content Management System (CMS) with a minimal set of functionalities. The application must satisfy the following requirements.

The CMS has a back-office (i.e., the administrative/management area) and a front-office (i.e., the version of the web application visible to everybody, without authentication).

In the back-office, the CMS allows any authenticated user to create and manage content (pages). Each page has the following properties:

- A title.
- An author (by default, the logged-in user).
- A creation date, i.e., the moment of creation.
- A publication date. According to this date, the page may be “draft” (empty publication date), “scheduled” (the publication date is in the future), or “published” (the publication date is today or in the past).
- Some blocks of content.

A block of content can be of three types: header, paragraph, or image. A page must have at least an header and at least one of the other two types of blocks. Images should be selected from a list of preloaded images (at least four different images must be available). Blocks can be ordered during the creation and the editing of the page (see the next paragraph). Header and paragraph blocks must contain text.

Each authenticated user, after log-in, can see a list of all the created pages (i.e., by any author) in a dedicated screen with the options to:

- Create a new page, by inserting all the needed properties, adding at least one header block, and at least one of the other blocks within the page. The creation date and the author cannot be changed. The blocks can be re-ordered at any time within the page during the creation process. How the re-ordering works is left to the student (e.g., arrows to move up and down the blocks). Blocks can also be removed, during the page creation process.
- Edit an existing page for which they are the author. By editing the page, the authenticated user can change all the properties (except the creation date and the author) and the content (including adding/removing and changing the order of blocks, but it must always contain at least an header and another type of blocks).
- Delete a page for which they are the author.

The CMS supports a special type of authenticated user, who is the admin of the entire application. There is no limit on the number of admins that can be present in the CMS. Any admin can perform all the operations of an authenticated user and, in addition:

- Edit or delete any page, i.e., even if they are not the author.
- Assign the authorship of a page to a different user.

- Commented [1]: Hello, I am asking this question to ... [1]
- Commented [2]: No. The user can create pages by ... [2]
- Commented [3]: Will there be any examinations be ... [3]
- Commented [4]: The oral discussion will start on th ... [4]
- Commented [5]: Do we need to have a general lan ... [5]
- Commented [6]: You need a login page, yes, simila ... [6]
- Commented [7]: When I login I can still see the fro ... [7]
- Commented [8]: When the doc introduces the front-o ... [8]
- Commented [9]: For the front office it is specified th ... [9]
- Commented [10]: No, up to you
- Commented [11]: Here, in the page, the author is “ ... [10]
- Commented [12]: Up to you, it's not specified
- Commented [13]: Thank you, another question: in ... [11]
- Commented [14]: can we display the username of ... [12]
- Commented [15]: @Salvatore: up to you. ... [13]
- Commented [16]: Is the author, during the creatio ... [14]
- Commented [17]: Always the creator
- Commented [18]: Can the creation date be set ma ... [15]
- Commented [19]: No, the creation date cannot be ... [16]
- Commented [20]: is there any condition on the ab ... [17]
- Commented [21]: No. As written below, “In the fro ... [18]
- Commented [22]: Thanks a lot, sorry I missed that 😊
- Commented [23]: If an user creates a page with publi ... [19]
- Commented [24]: The page status is not a “separ ... [20]
- Commented [25]: So, we don't need to show in th ... [21]
- Commented [26]: Status showing: it's not mandat ... [22]
- Commented [27]: Perfect, thanks!
- Commented [28]: Hello so, in general, the request ... [23]
- Commented [29]: Basically a simpler version of th ... [24]
- Commented [30]: Question 1: When the text spea ... [25]
- Commented [31]: The back-office is NOT the back ... [26]
- Commented [32]: Clearly, you will also need a bac ... [27]
- Commented [33]: so basically the back office is th ... [28]
- Commented [34]: Yea, same app, just different ... [29]
- Commented [35]: I' sorry I'm not sure I correctly ... [30]
- Commented [36]: The former. Keep in mind that t ... [31]
- Commented [37]: in the add page when I fill the ... [32]
- Commented [38]: Date
- Commented [39]: Could a page have multiple bloc ... [33]
- Commented [40]: A page can have multiple blocks, y ... [34]
- Commented [41]: is there a limit number of ... [35]
- Commented [42]: No limit. The author of the page ... [36]
- Commented [43]: Should the header text be the s ... [37]
- Commented [44]: A page can have multiple head ... [38]
- Commented [45]: I am sorry, I didn't understand v ... [39]
- Commented [46]: Yes, exactly. They should be in ... [40]
- Commented [47]: I can not visualize how images ... [41]
- Commented [48]: No. You have at least 4 images ... [42]
- Commented [49]: Should the pages displayed in t ... [43]
- Commented [50]: Not necessary
- Commented [51]: What does this mean ? ... [44]
- Commented [52]: after log-in, I see a list of all pages ... [45]
- Commented [53]: But the list of all pages in the wc ... [46]
- Commented [54]: yes, the difference is in the acti ... [47]
- Commented [55]: And the other difference is in wt ... [48]
- Commented [56]: So I can put part of the back-off ... [49]
- Commented [57]: but these two types of pages (d ... [50]
- Commented [58]: The common parts (all pages lis ... [51]

- Set up the **name of the website**, which must appear at the top of any screen (in the back-office and in the front-office).

In the front-office, instead, both authenticated and non-authenticated (anonymous) users will see the entire website with the defined name, a list of all *published* pages in chronological order (by publication date), and can read the full content of each page (and its properties).

The organization of these specifications in different screens (and possibly on different routes) is left to the student.

### Project requirements

- The application architecture and source code must be developed by adopting the best practices in software development, in particular those relevant to single-page applications (SPA) using React and HTTP APIs.
- The project must be implemented as a React application that interacts with an HTTP API implemented in Node+Express. The database must be stored in a SQLite file.
- The communication between client and server must follow the “two servers” pattern, by properly configuring CORS, and React must run in “development” mode with Strict Mode activated.
- The evaluation of the project will be carried out by navigating the application. Neither the behavior of the “refresh” button, nor the manual entering of a URL (except /) will be tested, and their behavior is undefined. Also, the application should never “reload” itself as a consequence of normal user operations.
- The root directory of the project must contain a README.md file, and have two subdirectories (client and server). The project must be started by running the two commands: “cd server; nodemon index.js” and “cd client; npm run dev”. A template for the project directories is already available in the exam repository. You may assume that nodemon is globally installed.
- The whole project must be submitted on GitHub, on the same repository created by GitHub Classroom.
- The project **must not include** the `node_modules` directories. They will be re-created by running the “npm install” command, right after “git clone”.
- The project may use popular and commonly adopted libraries (for example `day.js`, `react-bootstrap`, etc.), if applicable and useful. Such libraries must be correctly declared in the `package.json` file, so that the `npm install` command might install them.
- User authentication (login and logout) and API access must be implemented with `passport.js` and session cookies. The credentials should be stored in encrypted and salted form. The user registration procedure is not requested.

### Database requirements

- The project database must be implemented by the student, and must be pre-loaded with at least 4 users, with at least one who authored two pages, one who authored no pages, one who is an admin, and 2 pages per status (draft, published, programmed).

### Contents of the README.md file

The README.md file must contain the following information (a template is available in the project repository). Generally, each information should take no more than 1-2 lines.

Commented [93]: Is it ("name of the website") the "title"? If no, shouldn't the creator be able also to set up this parameter (otherwise they would be all blanks until an admin sets it)?

Let me know if I didn't explain myself or I misunderstood something. Thank you in advance.

Commented [94]: It's the "title" of the entire website. Like HeapOverrun or FilmLibrary (not the title of the single page, for which there is a separate property that can be set by each author). Until an admin set it up can ... [82]

Commented [95]: The name of the website should ... [83]

Commented [96]: Well, yes, given that it's a multi- ... [84]

Commented [97]: Hello. Is the "name of the websi ... [85]

Commented [98]: yes

Commented [99]: how many pages should we shc ... [86]

Commented [100]: It says "all the published pages" ... [87]

Commented [101]: with "all" do you mean all the ... [88]

Commented [102]: this sentence refers to the fron ... [89]

Commented [103]: Ascending or descending orde ... [90]

Commented [104]: It says "chronological order". It m ... [91]

Commented [105]: What is it meant by full conten ... [92]

Commented [106]: A logged-in user creates a new ... [93]

Commented [107]: In the front office do they need ... [94]

Commented [108]: They need to see a list of page ... [95]

Commented [109]: By clicking : could be a good s ... [96]

Commented [110]: It's up to you. The important th ... [97]

Commented [111]: I didn't understand which attrib ... [98]

Commented [112]: ciao, my question is can we us ... [99]

Commented [113]: No, you \*must\* use Express. ... [100]

Commented [114]: Thanks

Commented [115]: Can we use ORM libraries ... [101]

Commented [116]: If you know what they do and ... [102]

Commented [117]: is possible use other libraries ... [103]

Commented [118]: Yes, if you know/can explain ... [104]

Commented [119]: Since it is not required, does ... [105]

Commented [120]: Yes, do not implement registr ... [106]

Commented [121]: inside the database we have ... [107]

Commented [122]: It's pretty bad to store entire i ... [108]

Commented [123]: So the best choice is to save ... [109]

Commented [124]: I don't want/I cannot commen ... [110]

Commented [125]: let me stress that these image ... [111]

Commented [126]: Can we use our GitHub repo ... [112]

Commented [127]: Images should be in the proje ... [113]

1. Server-side:
  - a. A list of the HTTP APIs offered by the server, with a short description of the parameters and of the exchanged objects
  - b. A list of the database tables, with their purpose
2. Client-side:
  - a. A list of 'routes' for the React application, with a short description of the purpose of each route
  - b. A list of the main React components
3. Overall:
  - a. A screenshot of the screen for creating a new page and one for the screen with the list of all pages, both as a non-admin. These screenshots must be embedded in the README by linking two images committed in the repository.
  - b. Usernames and passwords of the users.

### Submission procedure

To correctly submit the project, you must:

- **Be enrolled** in the exam call.
- **Accept the invitation** on GitHub Classroom, and correctly **associate** your GitHub username with your student ID.
- **Push the project** in the **main branch** of the repository created for you by GitHub Classroom. The last commit (the one you wish to be evaluated) must be **tagged** with the tag **final** (note: final is all-lowercase, and it is a git 'tag', nor a 'commit message').

Note: to tag a commit, you may use (from the terminal) the following commands:

```
# ensure the latest version is committed
git commit -m "...comment..."
git push

# add the 'final' tag and push it
git tag final
git push origin --tags
```

Alternatively, you may insert the tag from GitHub's web interface (follow the link 'Create a new release').

To test your submission, these are the exact commands that the teachers will use to download and run the project. You may wish to test them on a clean directory:

```
git clone ...yourCloneURL...
cd ...yourProjectDir...
git pull origin main # just in case the default branch is not main
git checkout -b evaluation final # check out the version tagged with
'final' and create a new branch 'evaluation'
(cd client ; npm install; npm run dev)
(cd server ; npm install; nodemon index.js)
```

Ensure that all the needed packages are downloaded by the npm install commands. Be careful: if some packages are installed globally, on your computer, they might not be listed as dependencies. Always check it in a clean installation.

The project will be tested under Linux: be aware that Linux is case-sensitive for file names, while Windows and macOS are not. Double-check the case of import and require() statements.

**Commented [128]:** Excuse me, is it possible for a logged user to create 2 pages with the same attributes (title, author, creation/publication date)?

**Commented [129]:** It's not specified, so it's up to you to decide if and how to handle those cases

Page 1: [1] Commented [1] fatemeh maddahzadeh 6/5/23 9:49:00 AM

Hello, I am asking this question to be sure that I understood the project correctly. When we say that any authenticated user can create the content of the pages, we mean that the user can insert html content through a form and we have to save this html inside the database and load it?

Page 1: [2] Commented [2] Luigi De Russis 6/5/23 1:39:00 PM

No. The user can create pages by using blocks of different type (header, image, paragraph). They will then add/move such blocks and the content they will insert will be either text or image. They will not write HTML directly.

Page 1: [3] Commented [3] Anonymous 6/9/23 4:29:00 PM

Will there be any examinations before the 26 for those that are finished? Or the first will be 26?

Page 1: [4] Commented [4] Luigi De Russis 6/9/23 4:38:00 PM

The oral discussion will start on the 26th with a calendar shared by the teachers of each course (and different according to the course). You have until the deadline to double-check and refine everything.

Page 1: [5] Commented [5] Pegah Yarahmadi 6/6/23 5:27:00 PM

Do we need to have a general landing page that has the login and other stuff? Or any role directly sees the page with the related features?

Page 1: [6] Commented [6] Luigi De Russis 6/6/23 5:34:00 PM

You need a login page, yes, similarly to what we did in class/in the lab, with login info in the README (see towards the end of the doc)

Page 1: [7] Commented [7] filippo restori 6/7/23 2:53:00 PM

When I login I can still see the front-office?

Page 1: [8] Commented [8] Luigi De Russis 6/7/23 3:02:00 PM

When the doc introduces the front-office, it says: "In the front-office, instead, both authenticated and non-authenticated (anonymous) users...". Hence, yes

Page 1: [9] Commented [9] Margherita Lavena 6/12/23 4:37:00 PM

For the front office it is specified that pages should be ordered by publication date. Do we have to sort pages in a specific way in the back office?

Page 1: [10] Commented [11] Salvatore Rossetta 6/10/23 8:40:00 AM

Here, in the page, the author is "Name Surname" or we can use the email of the logged in user? If it's with name and surname, can we assume that the names are all different or we have to consider the cases in which some people may have the same name?

Page 1: [11] Commented [13] Salvatore Rossetta 6/10/23 2:47:00 PM

Thank you, another question: in the creation of a page we'll have a form to write title, publication date and blocks; it's up to us how to implement the "add blocks" or is there way we "have" to do it?

Page 1: [12] Commented [14] Lorenzo Fezza 6/10/23 2:51:00 PM

can we display the username of the author in the page or is it necessary to define a Name attribute in the DB?

Page 1: [13] Commented [15] Luigi De Russis 6/10/23 2:58:00 PM

@Salvatore: up to you.

@Lorenzo: up to you, the important thing is that there is an author.

Page 1: [14] Commented [16] Endri Sefa 6/9/23 5:19:00 PM

Is the author, during the creation of a page, always the creator or can also be inserted another user?

Page 1: [15] Commented [18] Lorenzo 6/7/23 5:08:00 PM

Can the creation date be set manually by the author (example if he wants to set it to the past)? Or we simply have to force the creation date to the real moment of creation (so when the user effectively creates it on our webpage)?

Page 1: [16] Commented [19] Fulvio Corno 6/8/23 11:27:00 AM

No, the creation date cannot be modified or set by the author (only the publication date may be customized)

Page 1: [17] Commented [20] Brendon Mendicino 6/8/23 4:03:00 PM

is there any condition on the ability of non-authenticated users to see "draft" pages or "scheduled" pages before the publish date? For example a "draft" page is uploaded, if I'm not an authenticated user will I be able to see it?

Page 1: [18] Commented [21] Luigi De Russis 6/9/23 7:46:00 AM

No. As written below, "In the front-office, instead, [...] non-authenticated (anonymous) users will see the entire website with the defined name, a list of all published pages [...]". Notice the **\*\*published\*\*** pages.

Page 1: [19] Commented [23] Mattia Chiarle 6/8/23 6:06:00 AM

If an user creates a page with publication date on the 9th of June for example, will the page status automatically change from scheduled to published on that date without any further user action?

Is it possible to update this field in the future, i.e. is it possible to change the page status from published to draft/scheduled once it has been published?

Page 1: [20] Commented [24] Fulvio Corno 6/8/23 11:29:00 AM

The page status is not a "separate" piece of information.

Simply put, based on the comparisons:

- publication date exists?
  - now >= publication date?
- the page will be visible or not

Page 1: [21] Commented [25] Mattia Chiarle 6/8/23 5:47:00 PM

So, we don't need to show in the back-office the status of each page, right?

Furthermore, if a publication date is set on a certain day, from that date the post should be visible (since it becomes published). Does this mean that, once the publication date is reached, it'll become the creation date and it won't be possible to further modify it?

Page 1: [22] Commented [26] Luigi De Russis 6/9/23 7:49:00 AM

Status showing: it's not mandatory.

Publication date: no, the creation date is the date in which first the page has been created. The publication date is the date in which the page has been published. They can contain the same date (if the page is published when it's created) but they are separate properties.

Page 1: [23] Commented [28] pedram nikjooy 6/7/23 1:21:00 PM

Hello so, in general, the requested project is basically delivering cms, and as a back office, it meant like admin part of WordPress or Joomla, or Drupal?

Page 1: [24] Commented [29] Luigi De Russis 6/7/23 2:16:00 PM

Basically a simpler version of those, yes. The back-office is defined exactly as “admin part”

Page 1: [25] Commented [30] nicola gasparotto 6/5/23 7:45:00 AM

Question 1: When the text speaks about the back office in the first lines, does it mean to implement the actions that are possible on the server side? ( the index.js file tested with test.http ? ) or have we to implement a further page for the actions and type of user back office?

Page 1: [26] Commented [31] Luigi De Russis 6/5/23 7:49:00 AM

The back-office is NOT the back-end. It's the part of the React app that act as an “administrative area” where authenticated users can perform their actions. Another part of the React app is the front-office, where everybody (including the non-authenticated users) can see the published content.

Page 1: [27] Commented [32] Luigi De Russis 6/5/23 7:50:00 AM

Clearly, you will also need a back-end in Express with the APIs etc. (as you've seen in class and experienced in the lab)

Page 1: [28] Commented [33] nicola gasparotto 6/5/23 7:57:00 AM

so basically the back office is the same react app, as back office we can intend the possibility to create, edit and delete a page?

Page 1: [29] Commented [34] Luigi De Russis 6/5/23 8:46:00 AM

Yea, same app, just different visualization/possible actions and options and accessible to authenticated users only

Page 1: [30] Commented [35] Alessia Leclercq 6/5/23 11:25:00 AM

I' sorry I'm not sure I correctly understood the difference. Are front-office and back-office something the logged-in user can choose to see (i.e., they can choose between the front-office of the back-office views) or are they the same page but with added functionalities in the case of the back-office?

Page 1: [31] Commented [36] Luigi De Russis 6/5/23 1:39:00 PM

The former. Keep in mind that the front-office is visible also to non-authenticated users (i.e., random visitors of the web application) and the back-office shows also the drafted and scheduled pages

Page 1: [32] Commented [37] Otabek Fayziev 6/10/23 4:50:00 PM

in the add page when I fill the publication date, it should select with these 3 options (draft, scheduled, published) or it should be exact date with input type date?

Page 1: [33] Commented [39] Simone Licitra 6/5/23 7:38:00 AM

Could a page have multiple blocks? I think so, but do they have an order when we have to show them?

Page 1: [34] Commented [40] Luigi De Russis 6/5/23 7:43:00 AM

A page can have multiple blocks, yes. The ones reported in the text are the minimum required (“at least...”). The order is specified by the authors (read below under “create a new page”)

Page 1: [35] Commented [41] Andres Tabima Romero 6/7/23 7:28:00 AM

is there a limit number of blocks content ?

Page 1: [36] Commented [42] Luigi De Russis 6/7/23 2:17:00 PM

No limit. The author of the page can have as many blocks as preferred in the page (within the requirements of at least 1 header block etc.)



Page 1: [37] Commented [43] Lorenzo 6/6/23 8:12:00 AM

Should the header text be the same as the page title?

Page 1: [38] Commented [44] Luigi De Russis 6/6/23 3:39:00 PM

A page can have multiple headers in the content. One is required. The title is a different property of the page.

Page 1: [39] Commented [45] Francesca Fusco 6/9/23 3:29:00 PM

I am sorry, I didn't understand where the images come from. Does the user have to choose from at least four different images "proposed" by us? In practice, do the images have to be (for example) in a folder of our project?

Page 1: [40] Commented [46] Luigi De Russis 6/9/23 4:17:00 PM

Yes, exactly. They should be in a folder of your project.

Page 1: [41] Commented [47] Otabek Fayziev 6/11/23 5:48:00 PM

I can not visualize how images should look. As I understood, we can choose from 0-4 images, among pre-loaded images and if I choose for example 2 images it will be one block. And for the same page I can choose another set of images from 0-4 among the same pre-loaded images and it will be the second block right?

By this logic For example can user choose 8 images from 4?

Page 1: [42] Commented [48] Luigi De Russis 6/11/23 6:14:00 PM

No. You have at least 4 images pre-loaded and available for the entire application. When a user creates a page and add an image block, he/she can choose 1 image for that block, among the set of pre-loaded images. If the user adds another image block to the same page, he/she will be able to choose 1 image among the same set of pre-loaded images.

Page 1: [43] Commented [49] Vanessa S. 6/11/23 9:10:00 PM

Should the pages displayed in the back-office be sorted chronologically like in the front-office, or is it not necessary?

Page 1: [44] Commented [51] Francesco Virga 6/5/23 3:11:00 PM

What does this mean ?

That the main page of the app changes if the user is logged in or that every authenticated user has their own content management page with a different url in which they see all the pages + their own?

Page 1: [45] Commented [52] Fulvio Corno 6/5/23 3:16:00 PM

after log-in, I see a list of all pages in the website (created by all authors)

It can be the same 'home' page that changes, or a different page accessible after login.

Page 1: [46] Commented [53] nicola gasparotto 6/5/23 3:19:00 PM

But the list of all pages in the website is not already visible for all the kind of users? authenticated and not? the only difference will be what kind of actions a 'user' can or cannot do, is it correct?

Page 1: [47] Commented [54] Fulvio Corno 6/5/23 3:21:00 PM

yes, the difference is in the actions that can be applied

Page 1: [48] Commented [55] Luigi De Russis 6/5/23 3:45:00 PM

And the other difference is in which pages are visible: draft and scheduled pages (with a date in



the future) won't be visible to non-logged-in users

**Page 1: [49] Commented [56] Francesco Virga 6/5/23 3:50:00 PM**

So I can put part of the back-office in the main page or I can put it on a completely different page right ?

**Page 1: [50] Commented [57] nicola gasparotto 6/5/23 3:51:00 PM**

but these two types of pages (draft and scheduled) are visible to ALL logged in users or only to the user who made them?

**Page 1: [51] Commented [58] Francesco Virga 6/5/23 3:53:00 PM**

The common parts (all pages list and edit/add/delete buttons) should be visible to all logged in users.

**Page 1: [52] Commented [59] Luigi De Russis 6/5/23 3:58:00 PM**

Draft/scheduled: all logged-in users. The logged-in users who don't authored those pages cannot do anything on those.

**Page 1: [53] Commented [60] Luigi De Russis 6/5/23 4:00:00 PM**

@Francesco Virga: given that the front-office is visible to both authenticated and non-authenticated users while the back-office is visible to logged-in users only, it would be easier to have them separated (they can optionally share some components, for instance)

**Page 1: [54] Commented [61] nicola gasparotto 6/5/23 4:04:00 PM**

Is possible to have a usercontext global variable and decide by means of it the properties of the element displayed and their accessibility? ( the possibility of the back office actions for the admins )

**Page 1: [55] Commented [62] Luigi De Russis 6/5/23 5:41:00 PM**

How do you implement the requirements is up to you, provided that the implementation is complying with the specifics and you can explain why/the pros/cons of your choice

**Page 1: [56] Commented [63] Luigi De Russis 6/5/23 5:42:00 PM**

Let me stress one more time that the front-office must be visualized by both the logged-in and the non-logged-in users in the same way (logged-in users will also have access/see the back-office, but this is another story)

**Page 1: [57] Commented [64] nicola gasparotto 6/7/23 12:23:00 PM**

is not so clear, so logged in user can choose wich one of the two: back office or front office view? so logged user can have the view with edit and delete button or a view without nothing and without drft pages like non logged user?

**Page 1: [58] Commented [65] Luigi De Russis 6/7/23 12:47:00 PM**

Yes. The front-office is accessible by both logged-in users and non logged-in. Logged-in users can also see (and operate) in the back-office.

**Page 1: [59] Commented [66] A C 6/7/23 1:01:00 PM**

Excuse me, you said that only logged-in users can see draft/scheduled pages. Can they see published pages in the same list too or do we have to keep these ones in a different list/route accessible to everybody?

**Page 1: [60] Commented [67] Luigi De Russis 6/7/23 1:14:00 PM**

The text says that all the pages (including draft/scheduled) are visible in the back-office to the logged-in users. In a dedicated screen. One screen in the back-office. As I suggested before, if you keep the back-office views separate

from the front-office it will be easier due to the many differences among the two (e.g., the fact that some pages are visible in one but not in the other, that logged-in users can access to both, etc.)

**Page 1: [61] Commented [68] Otabek Fayziev 6/11/23 4:36:00 PM**

by saying "blocks can be re-ordered at any time", does it mean only header and images and/or paragraph can be re-ordered?

and does re-order mean, let's say header, images and paragraph are separate containers, re-ordering means that we can change position of any containers right?

**Page 1: [62] Commented [69] Luigi De Russis 6/11/23 4:46:00 PM**

It means what you said: there will be blocks in a page (for example, 2 headers, 1 image, and 3 paragraphs) and those can be moved from their position at any time before clicking "save" and thus completing the creation of the page.

**Page 1: [63] Commented [70] Otabek Fayziev 6/11/23 5:00:00 PM**

should headers, images and paragraphs be separate container each and the contents only within these containers can be moved from their position, for example, 2 headers, 3 images, and 4 paragraphs, and i can only move the 2 headers only within the container headers, similarly i can move 3 images only within images container. Or can i move a header to the images' or paragraphs' section or the other way around, namely, they can all be mixed up?

**Page 1: [64] Commented [71] Luigi De Russis 6/11/23 5:11:00 PM**

They are separate blocks. Let's say that you create 1 header block, then add 1 image block, 1 paragraph block, another paragraph block, and 1 more header block. Then you should be able to move the last header block before the last paragraph and the image block after all the other blocks in the page.

**Page 1: [65] Commented [72] Simone Licitra 6/9/23 7:35:00 AM**

I can use react-beautiful-dnd for drag and drop the items of the table?

**Page 1: [66] Commented [73] Luigi De Russis 6/9/23 7:59:00 AM**

The overall suggestion is to keep things as simple as possible in your choices. In general, the project can use any library that are useful, if they do NOT solve a significant part of the project and you can explain what you did in your code with that plugin.

**Page 1: [67] Commented [74] Luigi De Russis 6/9/23 8:00:00 AM**

I checked the plugin and I see no problem in using it, keeping in mind what I wrote above

**Page 1: [68] Commented [76] Simone Licitra 6/9/23 4:31:00 PM**

Good afternoon, with "changing the order of blocks" we mean swap two item or re order the entire list of blocks?

N.B: react-beautiful-dnd didn't work, I wasted a day on this plugin. I advise to use dnd-kit.

**Page 1: [69] Commented [77] Luigi De Russis 6/9/23 4:40:00 PM**

It means moving one block in another position (up or down from the original position).

Re the "dnd" plugins: it's fine to do the ordering without drag and drop. A specific method is NOT required, the important thing is that blocks can be ordered in some ways. The text exemplifies with "arrows to move up and down the blocks", for instance.

**Page 1: [70] Commented [78] Francesco Virga 6/9/23 5:00:00 PM**

Is it necessary to be able to edit the type of a single block during the creation/editing of the page?

**Page 1: [71] Commented [79] Luigi De Russis 6/9/23 5:14:00 PM**

No, just adding, removing, and changing the order of blocks

**Page 1: [72] Commented [81] Alessio Cappello 6/6/23 10:40:00 AM**

Should we add an Edit page where we save all the updates made by the user at the end or it is done automatically when the user makes a change?

**Page 1: [73] Commented [82] Luigi De Russis 6/6/23 3:40:00 PM**

Keep it simple. A "save" button at the end is more than fine. Auto-saving isn't required.

**Page 1: [74] Commented [83] nicola gasparotto 6/5/23 4:29:00 PM**

admin user can change the date of the page or not? because logged in users cannot change date of the webpage. Also, by means of editing the logged user or admin can add and remove content of the page? more than one headers, paragraph and images?

Then, can everyone view in the specific webpage all the information about the page or not? creation date and publication date. Obviously if the page is draft or scheduled is clear that the not logged user cannot view the entire page, but normally the information in the page are:

- title
- author
- creation date and publication date
- blocks of contents

Is this correct?

**Page 1: [75] Commented [84] Luigi De Russis 6/6/23 3:46:00 PM**

Too many questions... let me know if I lost any:

- Admins can do everything that a logged-in user can do, but in all pages (not only those authored by him/her). They cannot change the \*creation date\* as such.
- Yes, editing means... editing. Users can change the content, add/remove blocks, re-order blocks, change the various properties (except the ones listed), etc.
- Everyone in the front-office can see all the page info (if the page is published), yes.

**Page 1: [76] Commented [85] Haochen He (ToomYBooM) 6/11/23 12:24:00 PM**

what does mean by assigning the authorship? if i am not wrong this should be like an admin could assign the authorship to a different user (different from the page author ) so that user can modify and delete?

**Page 1: [77] Commented [87] Haochen He (ToomYBooM) 6/11/23 5:37:00 PM**

since the requirement say assign that's mean i can just limit to assign it and not deleting it right?

**Page 1: [78] Commented [88] Luigi De Russis 6/11/23 6:11:00 PM**

An admin can assign the ownership of page to another user. Each page must have an author. What do you mean with "deleting"?

**Page 1: [79] Commented [89] Haochen He (ToomYBooM) 6/12/23 7:29:00 AM**

Sorry if i explained not so well by deleting i mean removing ,like as an admin I can assign so that user can see a page that the author is not him, by removing that user cannot see anymore

that page before I assigned authorship

Page 1: [80] Commented [90] Luigi De Russis 6/12/23 8:55:00 AM

When you assign the authorship to a new user (as an admin), you remove the authorship of the previous person. There is not a dedicate "remove function", but it a consequence of assigning it to another user.

Page 1: [81] Commented [91] Haochen He (ToomYBooM) 6/12/23 9:59:00 AM

So it's not possible to assign the authorship of a page to multiple users right?

Page 2: [82] Commented [94] Luigi De Russis 6/10/23 4:43:00 PM

It's the "title" of the entire website. Like HeapOverrun or FilmLibrary (not the title of the single page, for which there is a separate property that can be set by each author). Until an admin set it up can default to a default value, for instance.

Page 2: [83] Commented [95] Salvatore Rossetta 6/11/23 4:17:00 PM

The name of the website should be stored in a database to be the same the next time the application will be loaded or it can be just a variable (set to a default value) so that when the app is refreshed it gets back to this default value?

Page 2: [84] Commented [96] Luigi De Russis 6/11/23 4:44:00 PM

Well, yes, given that it's a multi-user application (with potentially multiple admins that can change the website name)

Page 2: [85] Commented [97] Alessia Leclercq 6/5/23 11:18:00 AM

Hello. Is the "name of the website" something like the "HeapOverrun" or the "FilmLibrary" we had in the class exercises/laboratories?

Page 2: [86] Commented [99] Otabek Fayziev 6/10/23 2:35:00 PM

how many pages should we show in the front office?

Page 2: [87] Commented [100] Luigi De Russis 6/10/23 2:59:00 PM

It says "all the published pages". So: all.

Page 2: [88] Commented [101] filippo restori 6/12/23 7:08:00 AM

with "all" do you mean all the published pages (so no draft/scheduled pages)? If yes, if a user create a page with publicationDate draft or scheduled, only him and the admin can see that page in the list of all pages?

Page 2: [89] Commented [102] Luigi De Russis 6/12/23 10:09:00 AM

this sentence refers to the front-office. In the front-office, all the published pages (no draft/scheduled) in chronological order.

In the back-office, instead, \*all\* logged-in users will see all the pages independently from their "status" (draft, scheduled, published). Clearly, only the authors+admin will be able to edit/delete their own pages.

Page 2: [90] Commented [103] Gianluca Alfieri 6/10/23 7:46:00 AM

Ascending or descending order?  
From latest to oldest or viceversa?

Page 2: [91] Commented [104] Luigi De Russis 6/10/23 2:28:00 PM

It says "chronological order". It means "in the order the events happened". So oldest to newest.

Page 2: [92] Commented [105] Jacobo Bou 6/5/23 5:30:00 PM

What is it meant by full content of the page? It is a specific route for each of the pages, with the

corresponding specific features?

Page 2: [93] Commented [106] Luigi De Russis 6/5/23 5:38:00 PM

A logged-in user creates a new page. Let's say with an header, two images, and three paragraphs in some order. When someone is looking at that single page on the front-office, they will see the full content: the header, the two images, and the three paragraphs, in the order defined by the page's author.

Page 2: [94] Commented [107] Leonardo Pavarino 6/6/23 10:00:00 AM

In the front office do they need to see a list of all pages with just title+author and then by clicking on one of them be brought to a separate route where they see all the contents of that page(images + text) or do they need to see all the pages (title + content) one after the other?

Page 2: [95] Commented [108] Luigi De Russis 6/6/23 3:48:00 PM

They need to see a list of pages in chronological order and, clicking on one of them, the full content of the page. The route organization is left to you.

Page 2: [96] Commented [109] Haochen He (ToomYBooM) 6/7/23 9:10:00 AM

By clicking : could be a good solution with a clicking a drop down button and then show all content of all page instead implementing the router solution to pass a dedicated page for each single page?

Page 2: [97] Commented [110] Luigi De Russis 6/7/23 12:50:00 PM

It's up to you. The important thing is to allow the various requirements: in this case, show the list of all published pages and, selecting one, see the full content of it.

Page 2: [98] Commented [111] Francesca Fusco 6/10/23 7:52:00 AM

I didn't understand which attributes of the page have to be displayed before clicking. Only the title, title and author, title and author and date...?

Thank you in advance

Page 2: [99] Commented [112] pedram nikjooy 6/12/23 2:36:00 PM

ciao, my question is can we use firebase for the server side, backend..

<https://firebase.google.com/docs/database/rest/start>

Page 2: [100] Commented [113] Luigi De Russis 6/12/23 2:46:00 PM

No, you \*must\* use Express, SQLite, with CORS, etc. exactly as we did during the course.

Page 2: [101] Commented [115] Filippo Rossi 6/14/23 7:55:00 AM

Can we use ORM libraries (like <https://github.com/prisma/prisma>) for database interactions?

Page 2: [102] Commented [116] Luigi De Russis 6/14/23 8:05:00 AM

If you know what they do and are able to explain, always using SQLite, yes

Page 2: [103] Commented [117] Haochen He (ToomYBooM) 6/5/23 12:51:00 PM

is possible use other libraries for frontend designs that the professor haven't cited in the course like antd

Page 2: [104] Commented [118] Luigi De Russis 6/5/23 1:42:00 PM

Yes, if you know/can explain how they work (in case they do something under the hood) and if they are similar to bootstrap, day.js, etc. That is, they help but don't replace entirely the goal of the work (e.g., a library that already does a good part of the CMS isn't acceptable).

Page 2: [105] Commented [119] Endri Sefa 6/11/23 5:41:00 PM

Since it is not required, does this mean that if we do not implement it, can we still get the

maximum points?

Page 2: [106] Commented [120] Luigi De Russis 6/11/23 6:12:00 PM

Yes, do not implement registration, just login/logout.

Page 2: [107] Commented [121] nicola gasparotto 6/5/23 8:46:00 AM

inside the database we have to store the information about all the pages, but within a page if we have n block of content that are images we have to store images inside the database. How can we do it? also is not clear from where the user can upload these images inside the page & database: he can choose from the pc? from a list of existing list inside a database?

Page 2: [108] Commented [122] Luigi De Russis 6/5/23 9:35:00 AM

It's pretty bad to store entire images within the DB. What you can do is store the name ("something.jpg" and the path maybe) of the image potentially involved in a block. Images should not be uploaded by the user: as reported in the text, they can choose from at least 4 pre-defined images from a list (defined by you, the developer). So you put the images somewhere and present them in a list so that the authenticated user can choose one of them for a block.

Page 2: [109] Commented [123] nicola gasparotto 6/5/23 9:45:00 AM

So the best choice is to save the images inside a directory in the client side, and on the server side save only the names of these images in the db? because i cannot view a possible solution to have the server delivers to the client these images through an api call. I think that the best solution is the first one proposed, to store these images in a directory inside the client dir. Is it correct?

Page 2: [110] Commented [124] Luigi De Russis 6/5/23 1:44:00 PM

I don't want/I cannot comment on specific solutions but, yes, storing the images in a directory and using them when needed it's a valid option

Page 2: [111] Commented [125] Fulvio Corno 6/5/23 1:53:00 PM

let me stress that these images are pre-defined (i.e., the admin can not change/upload them in any way)

Page 2: [112] Commented [126] Pegah Yarahmadi 6/8/23 7:58:00 PM

Can we use our GitHub repo to store the images and use their address in DB? Also, can we use a drag-drop plugin for ordering blocks or is it gonna make it so busy for the exam?

Page 2: [113] Commented [127] Luigi De Russis 6/9/23 7:51:00 AM

Images should be in the project (and uploaded in the repo for the evaluation). You can use drag and drop or any other ordering mechanism. You can use a plugin if you can explain how it's working in your code. The overall suggestion is always to keep things as simple as they can be.